Attorney Docket No.: 042390.P6942 *Patent*

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	
Nardin et al.	Examiner: Craig, Dwin M.
Application No.: 09/475,717	Art Unit: 2123
Filed: December 30, 1999	
For: METHOD AND APPARATUS FOR FULLY AUTOMATED SIGNAL INTEGRITY ANALYSIS FOR DOMINO CIRCUITRY	

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

DECLARATION PURSUANT TO 37 C.F.R. §1.131

Sir:

We, Mark D. Nardin, Hans Greub, and Sapumal Wijeratne, do hereby declare that:

1. We are the co-inventors of the above-captioned patent application and of the subject matter described and claimed therein.

2. Intel Corporation of Santa Clara, California, is the assignee of the patent application described above.

3. We are currently employed by Intel Corporation.

4. Prior to November 1, 1999, we jointly reduced to practice the invention as claimed in the above-captioned patent application (hereinafter "the present invention") in this country, as evidenced by Exhibits A, B, C, D, and E. All of these documents, in their unredacted form, were generated prior to November 1, 1999.

5. Exhibit A is a redacted Modification Log of software code, which practiced the present invention. The Modification Log shows that the present invention was reduced to practice prior to November 1, 1999. The last entry in the Modification Log was entered prior to November 1, 1999.

6. Exhibit B is a redacted portion of Parameter Extraction Code for extracting parameters of a set of domino logic circuits according to an embodiment of the present invention. The Parameter Extraction Code was reduced to practice prior to November 1, 1999.

7. Exhibit C is a redacted portion of an Output Log of software code that practiced the present invention. The Output Log was generated by the software code simulating a set of domino logic circuits prior to November 1, 1999.

8. Exhibit D is an unredacted Simulation Sequence File for simulating a set of domino logic circuits according to an ordered list. The Simulation Sequence File was generated by software code that practices the present invention prior to November 1, 1999.

9. Exhibit E is a redacted Simulation Time Stamp Log for a simulation executed on a set of domino logic circuits according to the Simulation Sequence File of Exhibit D. The Simulation Time Stamp Log was generated prior to November 1, 1999 by software code that practiced the present invention.

10. Evidence supporting that "extracting parameters of a set of domino logic circuits" was reduced to practice includes:

a. Exhibit B: This Exhibit illustrates a portion of Parameter Extraction Code for extracting parameters from domino logic circuits.

b. Exhibit C: This Exhibit of an Output Log implicitly provides evidence of extracting parameters of a set of domino logic circuits since the simulation which generated the Output Log could not have occurred without first extracting parameters of the set of domino logic circuits.

11. Evidence supporting that "simulating each domino logic circuit of a set of domino logic circuits" was reduced to practice includes:

a. Exhibit C, page 1, lines 31-35, 37-41, 45-50, etc.: These portions of Exhibit C illustrate the simulation results of domino logic circuits and therefore evidence that domino logic circuits were simulated.

12. Evidence supporting that "reporting results of the simulation indicating whether any of the domino logic circuit is likely to generate an erroneous output" was reduced to practice includes:

a. Exhibit C, page 2, lines 3 (for example): The portion "+0.048V DYNOUT" indicates that the particular domino circuit simulated has a positive noise margin

and therefore has a low likelihood to generate an erroneous output. A negative noise margin would be an indication that the particular domino circuit is likely to generate an erroneous output. The greater a negative noise margin the more likely an erroneous output. The more positive a noise margin the less likely an erroneous output.

b. Exhibit C, page 2, line 43: This portion of Exhibit C indicates that zero domino circuits of the simulated test circuit were likely to generate an erroneous output.

13. Evidence supporting that "scheduling a set of domino logic circuits into an ordered list, the ordered list positioning all domino logic circuits of the set of domino logic circuits feeding into an input of another domino logic circuit of the set of domino logic circuits before a position of the another domino logic circuit in the ordered list" was reduced to practice includes:

a. Exhibit D, page 1, lines 5, 13, 33, 43, and 50: The "simulation counts" schedule domino logic circuits into an ordered list or stages. This ordered list positions all domino logic circuits feeding into an input of another domino logic circuit before a position of the another domino logic circuit.

14. Evidence supporting that "simulating each domino logic circuit according to the ordered list" was reduced to practice includes:

a. Exhibit E: This Exhibit illustrates start and end time stamps of each stage of domino logic circuits scheduled for simulation in the Simulation Sequence File of Exhibit D. As can be seen the latest simulation end time of stage 1 "14:59:50" (Exhibit E, page 1, line 16) is before the earliest simulation start time of stage 3 "15:00:56" (Exhibit E, page 1, line 30). The latest simulation end time of stage 3 "15:47:53" (Exhibit E, page 1, line 40) is before the earliest simulation start time of stage 4 "15:49:02" (Exhibit E, page 1, line 50), and so on. Note, state 2 is not included in Exhibit E because stage 2 of the circuit under test did not include any logic circuits to simulate.

15. Evidence supporting that "determining whether any of the domino logic circuits is likely to generate an erroneous output" was reduced to practice includes:

a. Exhibit C, page 2, line 43: The indication that "0 domino circuits has negative noise margins" is a determination whether any of the domino logic circuits of the

circuit under test were likely to generate an erroneous output. Any domino logic circuit having a negative noise margin would have been likely to generate an erroneous output.

We hereby declare that all statements made herein of our own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the above-identified application or any patent issued thereon.

Respectfully submitted,

Date April 15, 2004 _____
Mark D. Nardin

Date April 14, 2004 _____
Hans Greub

Date APRIL 14, 2004 _____
Sapumal Wijeratne

EXHIBIT A

(Modification Log)

```
1  #*************************************************************************
2  #* Filename: domino_manager      Project: Cop
3  #*************************************************************************
4  #*
5  #* (C) Copyright Intel Corporation, [redacted]
6  #* Licensed material -- Program property of Intel Corporation
7  #* All Rights Reserved
8  #*
9  #* This program is the property of Intel Corporation and is furnished
10 #* pursuant to a written license agreement. It may not be used, reproduced,
11 #* or disclosed to others except in accordance with the terms and conditions
12 #* of that agreement.
13 #*
14 #*************************************************************************
15 #*
16 #* Original Author: Hans J. Greub  Email: [redacted]
17 #*
18 #* Functional description:
19 #*
20 #* This script extracts domino circuits and simulates the dominos and
21 #* inverting gates igates in stages using dominosim for simulating the
22 #* the dominos for chargesharing, residual (propagated noise from the
23 #* input to the output), and the injected crosstalk voltage at the output,
24 #* and using go_nm to characterize UGNMH vs Vout for custom or zgcells
25 #* connected to dominos and and then propagates the worst case
26 #* voltage drop on the domino output through the inverting gates to get
27 #* the input residual for the next domino stage.
28 #* All propagated residuals are captured in the file:
29 #* xcap/domino/data/<fub>.residual
30 #* A margin report for all domino outputs is written to the file:
31 #* xcap/report/<fub>.domino_finalreport
32 #*
33 #*************************************************************************
34 #*
35 #*
36 #*************************************************************************
37 # Implementation Notes:
38 #*************************************************************************
39 #
40 # Data Structures
41 #
42 # The Domino Output Noise Info is stored in the hash:
43 # $DomOutput{$pathmill_node_name}=\@domino_output_record;
44 # each entry contains pointer to a domino_output_record with the following format:
45 #
46 @domino_output_record=($Reff,$Rline,$Ctot,$Cx,$Residual,$Peak,$Fub_Pin,$Supply_Noise,$ChargeSh
47 aring,$Average_Attacker_Slope,$assumed_fixed_value);
48 # The Domino Input Noise Info is stored in the hash:
49 # $DomInput{$pathmill_node_name}=\@domino_input_record;
50 # Each entry points to a record which contains:
51 #
52 @domino_input_record=($Reff,$Rline,$Ctot,$Cx,$Residual,$Source_of_Residual,$Peak,$Fub_Pin,$Suppl
53 y_Noise,$Average_Attacker_Slope);
54 # changed keys from ipath to pathmill notation
```

```
1   # - added the mapping hashes for simulation
2   #  %map_out2igate{$node} ="igate${id}$fub"
3   #  %map_out2domino{$node}="dom${id}$fub"
4   #  these hashes map an output node to a domino or igate cell name
5   # added the following hashes
6   #
7   @receiver_record=($domino_driven_input_pin,$source_config,$invelm_output,$invelm_name,$invelm_ty
8   pe);
9   # The hash %map_igate_out2igate_record maps igate outputs to igate records
10  # @igate_record=($invelm_type,$source_config,\@domino_driven_input_list,
11  #           \@domino_driven_input_pin_list,$invelm_name);
12
13  # obsolete $map_igate_receiver{$domino_driven_input}=\@receiver_records;
14  # $map_igate_out2cell_type{$igate_output}=$cell_type;
15  # The residual on igate outputs must be propagated thru
16  # passgates. The hash %short_igate2dynin with key $igate_output_node
17  # points to an array (list) of dynin nodes to which the residual
18  # needs to be propagated.
19  # $short_igate2dynin{$igate_node}=\@dynin_node_list;
20  # push(@{$short_igate2dynin{$igate_node}},$dynin_node);
21  #
22  # Modification Log
23  # [redacted] - added fub_boundary condition check for fub outputs
24  #       - added fub_boundary statements for fub input
25  #       - changed no receivers found on domino outputs to
26  #         warning messages to handle nocons better
27  # [redacted] - moving databases instead of deleting them!
28  # [redacted] - fixed bug in domino_stageN.pN cell list generation
29  # [redacted] - changed noise propagation from DYNOUT based to igate cell
30  #         based to conform to order in sim_sequence
31  # [redacted] - added -use_previous_results feature
32  # [redacted] - added archiving and output of $fub.residuals
33  # [redacted] - added database migration for -start_fresh option
34  # [redacted] - added $ENV{CSEJOBNOMAIL}="TRUE";
35  # [redacted] - removed path to /usr/home1/hgreub version of
36  #         igate_identify
37  # [redacted] - removed path checking for domino_extract because
38  #         it hangs in CTM
39  #       - changed tcsh path since /bin/tcsh does not work in
40  #         CTM
41  # [redacted] - fixed bug in migrate_dp which cause domino_manager
42  #         to quit if -start_fresh option is used and no db
43  #         datafiles exist
44  # [redacted] - fixed 'nbq -Pcs' instead $command_prefix bug in
45  #         domino simulate section
46  # [redacted] - added -f flag to tcsh to fix some problems with
47  #         setup in CTM
48  # [redacted] - added support for custom cells that the user wants
49  #         to treat like standard cells
50  #         if a cell custom_cell that is listed in the inv_element_fub.dat
51  #         file and thus was declared to be treated like a standard cell
52  #         in the xcap/domino/igate_no_extract_fub.dat file, domino_manager
53  #         looks for a command file "custom_cell.cmd" and if it exists
54  #         will simulate this cell once and read in the results
```

```
1   # [redacted] - removed -x from tcsh -f -x
2   # [redacted] - changed pathmill2plus to not add fubname prefix for fub pins
3   # [redacted] - changed read_transgate_domino_sim, looks like header in the
4   #       file changed
5   # [redacted] - fixed bug in UGNMH computation, lowest UGNMH instead of highest
6   #       UGNMH with lowest NT was kept
7   # [redacted] - fixed bug in residual propagation through passgates, the new
8   #       residual value was copied in without checking whether the existing
9   #       value is (worst case)
10  #     - fixed argument processing so that domino_manager -<anything> gives
11  #       usage message
12  # [redacted] - added handling of case if 2*($vout-2*$vout2) is zero
13  #       in compute_propagated_residual()
14  # [redacted] - added check for TIM version 2.8.b1
15  # [redacted] - added message to re-run xcap_mutex and xcap_change_psn
16  # [redacted] - added handling of domino/igate not reported condition in
17  #       sim seq file
18  # [redacted] - fixed migrate_db() for igate
19  # [redacted] - fixed worst noise level reported in domino_finalreport
20  # [redacted] - added an enhancement to deal with multiple tri-state drivers
21  #       connected to an igate output node (works for stdcells only)
22  # [redacted] - fixed domino residual propagation bug, fub.residuals was correct
23  #       but %DomInput data was still bad
24  # [redacted] - increased min chunk from 12 to 24 because of netbatch overflow.
25  # [redacted] - changed initial values in DomOutput to make sure dominos that
26  #       have not been simulated will fail
27  # [redacted] - added sanity checks to read_sim_seq files
28  # [redacted] - fixed residual propagation through passgates
29  # [redacted] - added archiving of siminofo file used for domino simulation
30
31
32  $VERSION="2.0";
33  $last_modified="[redacted]";
34  .
35  .
36  .
37
38                  This gives a time date of the LAST modification of some other "underlying" scripts
39                  that domino_manager calls to do needed functions.
40  [redacted]>ls -l
41  total 124
42  -rwxr-xr-x  1 [redacted] users  2151 [redacted] build_for_xcap
43  -rwxr-xr-x  1 [redacted] users  13009 [redacted] domino2ipath
44  -rwxr-xr-x  1 [redacted] users  10042 [redacted] domino_extract
45  -rwxr-xr-x  1 [redacted] users  7072 [redacted] ggate_extract
46  -rwxr-xr-x  1 [redacted] users  7710 [redacted] igate_extract
```

EXHIBIT B

(Parameter Extraction Code)

From the code "domino_extract":

```
#!/bin/csh

# Created [redacted] by Mark Nardin
# For use in extracting domino circuit netlists for simulation

set DOM_EXTRACT_EXE = $0

if ( ($#argv == 0) | ($1 == "-help") ) then
  echo " "
  echo "This MUST be run from a setup window where plus can be run. "
  echo " "
  echo " "
  awk '/^#BEGINhelp_message/ {\
    getline\
    while ( $1 != "#ENDhelp_message") {\
      print\
      getline\
    } }' $DOM_EXTRACT_EXE
  exit 0
endif

setenv WARD $WORK_AREA_ROOT_DIR
setenv FUB $1$2
setenv fub $1

if !(-e $WARD/plus/frz/xcap_$fub.frz) then
  echo " "
  echo " Can not find the required freeze file:"
  echo "    "$WARD/plus/frz/xcap_$fub.frz
  echo " "
  echo " Run the script: build_for_xcap "
  echo " "
  exit 0
endif


# Record the current directory
set CUR_DIR = `pwd`

# Make the master command file that needs to be executed in plus
#
rm -f $WARD/plus/cmd/domcall_tmp_$FUB.cmd
#
# Making the start-up sequence for PLUS to run
#
echo " Running plus and restarting the freeze file from xcap_<fub>.frz"
echo "restart xcap_$fub" > $WARD/plus/cmd/domcall_tmp_$FUB.cmd
#
# Making the series of commands that need to be run for each of the
# individual domino nodes
#
awk ' /^/ { \
```

```
1   print "put n "$1" domoutput_erc := TRUE"; \
2   print "@"'$WARD"'/plus/cmd/domselect_plus_"'$FUB"'.cmd"; \
3   print "@"'$WARD"'/plus/cmd/select_temp_"'$FUB"'.cmd"; \
4   print "system date"; \
5   print "simulate -nojob -ignore -selected -sdp dom"$2"ext"'$fub'""; \
6   print "system process_ext dom"$2"ext"'$fub"'.sdp -create_template"; \
7   print "system source "'$WARD"'/plus/cmd/make_delete_file_"'$FUB"'.tmp"; \
8   print "@"'$WARD"'/plus/cmd/delete_sources_"'$FUB"'.tmp" } ' \
9    $WARD/plus/erc/domout_nodes_$FUB.dat >> $WARD/plus/cmd/domcall_tmp_$FUB.cmd
10
11  # Make the plus command file that actually extracts the iPath
12  # command file statements
13  #
```

EXHIBIT C

(Output Log)

```
ptdl:[redacted]n>ls -l
total 528
-rwxr-xr-x  1 [redacted] cop     1139 [redacted] ##[redacted]-10:38:37#.ptdis91.gz
-rwxr-xr-x  1 [redacted] cop      473 [redacted] ##[redacted]-12:48:42#.ptdis12.gz
-rw-r--r--  1 [redacted] cop      265 [redacted] faaddc.domino_extract_audit.gz
-rw-r--r--  1 [redacted] cop     5749 [redacted] faaddc.domino_finalreport.complete.gz
-rw-r--r--  1 [redacted] cop     5759 [redacted] faaddc.domino_finalreport.gz
-rw-r--r--  1 [redacted] cop     5749 [redacted] faaddc.domino_finalreport.previous.gz
-rw-r--r--  1 [redacted] cop     3415 [redacted] faaddc.domino_simulate.audit.gz
-rw-r--r--  1 [redacted] cop   495820 [redacted] faaddc.xcap_finalreport.gz

ptdl:mnardin>gzless faaddc.domino_finalreport.gz
****************************
* DOMINO FLOW XCAP REPORT *
****************************

domino_manager version 2.0, last modified on [redacted]

Command Line    : domino_manager faaddc -simulate -parallel 8 -netbatch iss_short
TimeStamp       : [redacted]

USER            : [redacted]
WORK_AREA_ROOT_DIR: /prj/cop/work_root/feu/[redacted]/faaddc
Note: The worst domino input residual reported is the worst residual
propagated to the inputs from a previous domino stage, the worst case
domino input noise is the worst total noise (power_supply_noise+residual+xtalk)
on any domino input (not necessarily the input that had the worst residual)

Report for all DYNOUT Nodes sorted based on margin

 ?.???V DYNOUT faaddd/i34/pp[71]       (dom194faaddc)
      -W- no receiver found, verify NOCON!
      Voltage Drop: 0.130V (ChgSh(0.010V)+Residual(0.040V)+XTalk(0.055V)+PSN(0.025V))
      worst domino input noise   : 0.111V on node: faaddd/i34/i13/i1/pp2nn[3]
      worst domino input residual: 0.029V from dom245faaddc

 ?.???V DYNOUT faaddd/i34/gg[71]       (dom144faaddc)
      -W- no receiver found, verify NOCON!
      Voltage Drop: 0.199V (ChgSh(0.001V)+Residual(0.032V)+XTalk(0.141V)+PSN(0.025V))
      worst domino input noise   : 0.120V on node: faaddd/i34/i13/i1/gg2nn[1]
      worst domino input residual: 0.029V from dom245faaddc

*** The Noise on the following Domino Output Nodes is below the Receiver UGNMH ***

 +0.032V DYNOUT faaddd/i34/i31/gout[5]  (dom104faaddc)
      Voltage Drop: 0.186V (ChgSh(0.001V)+Residual(0.085V)+XTalk(0.075V)+PSN(0.025V))
      worst receiver UGNMH      : 1.582V (NT:0.218V) from
 zgca2nox800040x4000040x1024040x4000040
      worst domino input noise   : 0.197V on node: faaddd/i34/i31/gg2nn[1]
      worst domino input residual: 0.073V from dom55faaddc

 +0.037V DYNOUT faaddd/i34/gg[29]      (dom82faaddc)
      Voltage Drop: 0.208V (ChgSh(0.000V)+Residual(0.031V)+XTalk(0.152V)+PSN(0.025V))
      worst receiver UGNMH      : 1.555V (NT:0.245V) from
 zgca2nox1000040x4000040x1024040x4000040
      worst domino input noise   : 0.120V on node: faaddd/i34/i6/i1/gg2nn[1]
```

```
1       worst domino input residual: 0.028V from dom137faaddc
2
3   +0.048V DYNOUT faaddd/i34/gg[17]      (dom211faaddc)
4       Voltage Drop: 0.197V (ChgSh(0.000V)+Residual(0.031V)+XTalk(0.141V)+PSN(0.025V))
5       worst receiver UGNMH      : 1.555V (NT:0.245V) from
6   zgca2nox1000040x4000040x1024040x4000040
7       worst domino input noise  : 0.120V on node: faaddd/i34/i4/i1/gg2nn[1]
8       worst domino input residual: 0.028V from dom72faaddc
9
10  +0.050V DYNOUT faaddd/i34/pp[11]      (dom55faaddc)
11      Voltage Drop: 0.261V (ChgSh(0.010V)+Residual(0.031V)+XTalk(0.195V)+PSN(0.025V))
12      worst receiver UGNMH      : 1.489V (NT:0.311V) from
13  zgca2nox1400040x3600040x1024040x3600040
14      worst domino input noise  : 0.111V on node: faaddd/i34/i3/i1/pp2nn[3]
15      worst domino input residual: 0.028V from dom168faaddc
16
17  +0.051V DYNOUT faaddd/i34/pp[23]      (dom189faaddc)
18      Voltage Drop: 0.194V (ChgSh(0.010V)+Residual(0.031V)+XTalk(0.128V)+PSN(0.025V))
19      worst receiver UGNMH      : 1.555V (NT:0.245V) from
20  zgca2nox1000040x4000040x1024040x4000040
21      worst domino input noise  : 0.111V on node: faaddd/i34/i5/i1/pp2nn[3]
22      worst domino input residual: 0.028V from dom233faaddc
23
24  +0.055V DYNOUT faaddd/i34/pp[53]      (dom126faaddc)
25      Voltage Drop: 0.242V (ChgSh(0.010V)+Residual(0.033V)+XTalk(0.174V)+PSN(0.025V))
26      worst receiver UGNMH      : 1.503V (NT:0.297V) from zi0bna02he
27      worst domino input noise  : 0.111V on node: faaddd/i34/i10/i1/pp2nn[3]
28      worst domino input residual: 0.029V from dom24faaddc
29
30  .
31  .
32  .
33
34  ****************************
35  * SUMMARY of DOMINO REPORT *
36  ****************************
37
38  249 dominos were found in FUB: faaddc
39
40  0 dominos were not mapped or extracted
41  2 dominos had no receivers (NOCONS?)
42  0 dominos were assumed to be fixed for noise propagation
43  0 domino circuits had negative noise margins
```

EXHIBIT D

(Simulation Sequence File)

```
1   sim_seq_faaddc.dat:
2   #<node_type>        <node_name>
3   ----------------------------------------------------------------------------------------------------
4
5   #simulation_count    1.000
6   domino_node         faaddd{p62faaddd}/i34{p62faadyn72add}/i10{p62faadnew2zi0madd_add6c}%g[0]
7   domino_node         faaddd{p62faaddd}/i34{p62faadyn72add}/i10{p62faadnew2zi0madd_add6c}%p[0]
8   domino_node         faaddd{p62faaddd}/i34{p62faadyn72add}/i10{p62faadnew2zi0madd_add6c}%g[1]
9   domino_node         faaddd{p62faaddd}/i34{p62faadyn72add}/i10{p62faadnew2zi0madd_add6c}%p[1]
10  ..
11  ..
12  domino_node         faaddd{p62faaddd}/i34{p62faadyn72add}/i9{p62faadnew2zi0madd_add6c}%p[5]
13  #simulation_count    2.000
14  igate_node          faaddd{p62faaddd}/i34{p62faadyn72add}%qnn[48]
15  igate_node          faaddd{p62faaddd}/i34{p62faadyn72add}%qnn[49]
16  igate_node          faaddd{p62faaddd}/i34{p62faadyn72add}%qnn[50]
17  igate_node          faaddd{p62faaddd}/i34{p62faadyn72add}%qnn[51]
18  igate_node
19  faaddd{p62faaddd}/i34{p62faadyn72add}/i10{p62faadnew2zi0madd_add6c}/i0[1]{p62faaddczi0madd_pg
20  genc}%net100
21  igate_node
22  faaddd{p62faaddd}/i34{p62faadyn72add}/i10{p62faadnew2zi0madd_add6c}/i0[2]{p62faaddczi0madd_pg
23  genc}%net100
24  igate_node
25  faaddd{p62faaddd}/i34{p62faadyn72add}/i10{p62faadnew2zi0madd_add6c}/i0[3]{p62faaddczi0madd_pg
26  genc}%net100
27  igate_node
28  faaddd{p62faaddd}/i34{p62faadyn72add}/i10{p62faadnew2zi0madd_add6c}/i0[4]{p62faaddczi0madd_pg
29  genc}%net100
30  igate_node
31  faaddd{p62faaddd}/i34{p62faadyn72add}/i9{p62faadnew2zi0madd_add6c}/i1{p62fazi0madd_cla6c}%pp
32  2nn[1]
33  #simulation_count    3.000
34  domino_node         faaddd{p62faaddd}/i34{p62faadyn72add}%gg[50]
35  domino_node         faaddd{p62faaddd}/i34{p62faadyn72add}%gg[51]
36  domino_node         faaddd{p62faaddd}/i34{p62faadyn72add}%gg[52]
37  domino_node         faaddd{p62faaddd}/i34{p62faadyn72add}%gg[53]
38  ..
39  ..
40  domino_node         faaddd{p62faaddd}/i34{p62faadyn72add}%pp[50]
41  domino_node         faaddd{p62faaddd}/i34{p62faadyn72add}%gg[47]
42  domino_node         faaddd{p62faaddd}/i34{p62faadyn72add}%pp[47]
43  #simulation_count    4.000
44  igate_node          faaddd{p62faaddd}/i34{p62faadyn72add}/i16[3]{zi0madd_sume}%n0
45  igate_node          faaddd{p62faaddd}/i34{p62faadyn72add}/i16[3]{zi0madd_sume}%ggnn
46  igate_node          faaddd{p62faaddd}/i34{p62faadyn72add}/i31{p62faa2ndcla}%pp2nn[10]
47  ..
48  ..
49  igate_node          faaddd{p62faaddd}/i34{p62faadyn72add}/i31{p62faa2ndcla}%pp2nn[7]
50  #simulation_count    5.000
51  domino_node         faaddd{p62faaddd}/i34{p62faadyn72add}/i31{p62faa2ndcla}%pp[10]
52  domino_node         faaddd{p62faaddd}/i34{p62faadyn72add}/i31{p62faa2ndcla}%gout[5]
53  domino_node         faaddd{p62faaddd}/i34{p62faadyn72add}/i31{p62faa2ndcla}%gp[10]
54  ..
```

```
1   ..
2   domino_node          faaddd{p62faaddd}/i34{p62faadyn72add}/i31{p62faa2ndcla}%pp[7]
3   domino_node          faaddd{p62faaddd}/i34{p62faadyn72add}/i31{p62faa2ndcla}%gp[3]
4   #simulation_count     6.000
5   igate_node          faaddd{p62faaddd}/i34{p62faadyn72add}%coutnn[23]
6   igate_node          faaddd{p62faaddd}/i34{p62faadyn72add}%coutnn[29]
7   igate_node          faaddd{p62faaddd}/i34{p62faadyn72add}%coutnn[35]
8   igate_node          faaddd{p62faaddd}/i34{p62faadyn72add}%coutnn[41]
9   igate_node          faaddd{p62faaddd}/i34{p62faadyn72add}%coutnn[47]
10  igate_node          faaddd{p62faaddd}/i34{p62faadyn72add}%coutnn[53]
11  igate_node          faaddd{p62faaddd}/i34{p62faadyn72add}%coutnn[59]
12  igate_node          faaddd{p62faaddd}/i34{p62faadyn72add}%coutnn[65]
```

EXHIBIT E

(Simulation Time Stamp Log)

```
1  #[redacted]-14:10:39#.ptdis86:
2   | Starting time  : Fri [redacted] 14:10:39 [redacted]
3   | Command       : xcap/domino/data/nbq_domino_stage1.p1
4   | Finishing time : Fri [redacted] 14:41:59 [redacted]
5   ##[redacted]14:10:40#.ptdis77:
6   | Starting time  : Fri [redacted] 14:10:40 [redacted]
7   | Command       : xcap/domino/data/nbq_domino_stage1.p2
8   | Finishing time : Fri [redacted] 14:43:12 [redacted]
9   ##[redacted]-14:10:41#.ptdis97:
10  | Starting time  : Fri [redacted] 14:10:41 [redacted]
11  | Command       : xcap/domino/data/nbq_domino_stage1.p3
12  | Finishing time : Fri [redacted] 14:42:03 [redacted]
13  ##[redacted]-14:10:41#.ptdis75:
14  | Starting time  : Fri [redacted] 14:10:41 [redacted]
15  | Command       : xcap/domino/data/nbq_domino_stage1.p4
16  | Finishing time : Fri [redacted] 14:59:50 [redacted]
17  ##[redacted]-14:10:42#.ptdis116:
18  | Starting time  : Fri [redacted] 14:10:42 [redacted]
19  | Command       : xcap/domino/data/nbq_domino_stage1.p5
20  | Finishing time : Fri [redacted] 14:58:43 [redacted]
21  ##[redacted]-14:10:42#.ptdis108:
22  | Starting time  : Fri [redacted] 14:10:42 [redacted]
23  | Command       : xcap/domino/data/nbq_domino_stage1.p6
24  | Finishing time : Fri [redacted] 15:00:26 [redacted]
25  ##[redacted]-14:10:43#.ptdis14:
26  | Starting time  : Fri [redacted] 14:10:43 [redacted]
27  | Command       : xcap/domino/data/nbq_domino_stage1.p7
28  | Finishing time : Fri [redacted] 14:52:31 [redacted]
29  ##[redacted]-15:00:56#.ptdis78:
30  | Starting time  : Fri [redacted] 15:00:56 [redacted]
31  | Command       : xcap/domino/data/nbq_domino_stage3.p1
32  | Finishing time : Fri [redacted] 15:48:19 [redacted]
33  ##[redacted]-15:00:57#.ptdis99:
34  | Starting time  : Fri [redacted] 15:00:57 [redacted]
35  | Command       : xcap/domino/data/nbq_domino_stage3.p2
36  | Finishing time : Fri [redacted] 15:47:51 [redacted]
37  ##[redacted]-15:00:57#.ptdis109:
38  | Starting time  : Fri [redacted] 15:00:57 [redacted]
39  | Command       : xcap/domino/data/nbq_domino_stage3.p3
40  | Finishing time : Fri [redacted] 15:47:53 [redacted]
41  ##[redacted]-15:00:58#.ptdis89:
42  | Starting time  : Fri [redacted] 15:00:58 [redacted]
43  | Command       : xcap/domino/data/nbq_domino_stage3.p4
44  | Finishing time : Fri [redacted] 15:47:51 [redacted]
45  ##[redacted]-15:00:58#.ptdis87:
46  | Starting time  : Fri [redacted] 15:00:58 [redacted]
47  | Command       : xcap/domino/data/nbq_domino_stage3.p5
48  | Finishing time : Fri [redacted] 15:39:01 [redacted]
49  ##[redacted]-15:49:02#.ptdis97:
50  | Starting time  : Fri [redacted] 15:49:02 [redacted]
51  | Command       : xcap/domino/data/nbq_igate_stage4.p1
52  | Finishing time : Fri [redacted] 16:00:56 [redacted]
53  ##[redacted]-16:01:05#.ptdis86:
54  | Starting time  : Fri [redacted] 16:01:05 [redacted]
```

1 | Command : xcap/domino/data/nbq_domino_stage5.pl
2 | Finishing time : Fri [redacted] 16:14:30 [redacted]
3 ##[redacted]-16:15:07#.ptdis86:
4 | Starting time : Fri [redacted] 16:15:07 [redacted]
5 | Command : xcap/domino/data/nbq_igate_stage6.pl
6 | Finishing time : Fri [redacted] 16:21:12 [redacted]